

(19) 世界知的所有権機関
国際事務局



(43) 国際公開日
2005 年 8 月 25 日 (25.08.2005)

PCT

(10) 国際公開番号
WO 2005/078736 A1

- (51) 国際特許分類⁷: G11C 29/00, G01R 31/28
- (21) 国際出願番号: PCT/JP2004/001805
- (22) 国際出願日: 2004 年 2 月 18 日 (18.02.2004)
- (25) 国際出願の言語: 日本語
- (26) 国際公開の言語: 日本語
- (71) 出願人 (米国を除く全ての指定国について): 株式会社アドバンテスト (ADVANTEST CORPORATION) [JP/JP]; 〒1790071 東京都練馬区旭町1丁目32番1号 Tokyo (JP).
- (72) 発明者; および
- (75) 発明者/出願人 (米国についてのみ): 大川 和祥

(OKAWA, Kazuyoshi) [JP/JP]; 〒1790071 東京都練馬区旭町1丁目32番1号 株式会社アドバンテスト内 Tokyo (JP). 荻野 純子 (OGINO, Junko) [JP/JP]; 〒1790071 東京都練馬区旭町1丁目32番1号 株式会社アドバンテスト内 Tokyo (JP). 吉永 真之 (YOSHINAGA, Masayuki) [JP/JP]; 〒1790071 東京都練馬区旭町1丁目32番1号 株式会社アドバンテスト内 Tokyo (JP). 本多元 (HONDA, Hajime) [JP/JP]; 〒1790071 東京都練馬区旭町1丁目32番1号 株式会社アドバンテスト内 Tokyo (JP).

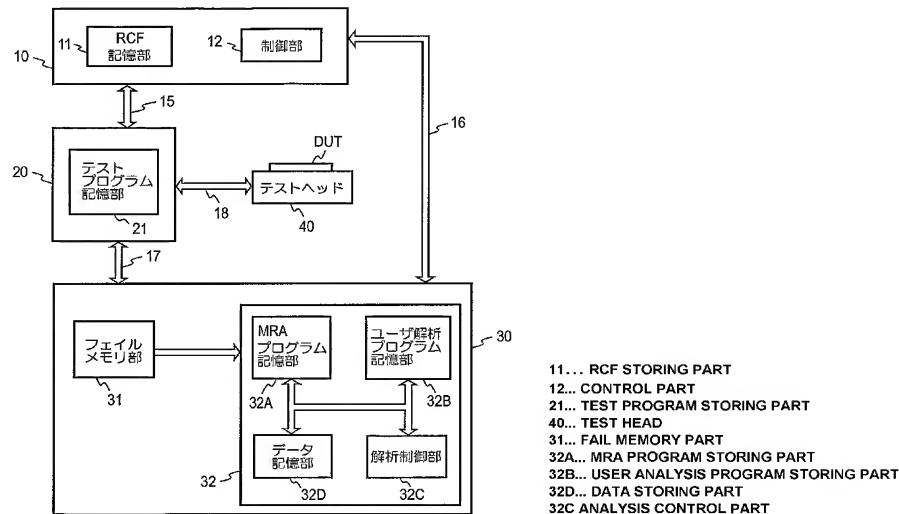
(74) 代理人: 草野 卓, 外 (KUSANO, Takashi et al.); 〒1600022 東京都新宿区新宿三丁目1番22号 新宿NSOビル4階 Tokyo (JP).

(81) 指定国 (表示のない限り、全ての種類の国内保護が可能): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR,

[続葉有]

(54) Title: SEMICONDUCTOR DEVICE TESTING APPARATUS AND TESTING METHOD

(54) 発明の名称: 半導体デバイス試験装置及び試験方法



(57) Abstract: A semiconductor device testing apparatus wherein a test processor (20) is used to apply a test signal to a DUT including a semiconductor memory, a failure/no failure test of the memory is performed based on the response signal, and wherein a remedy analysis dedicated calculation unit (30) is used to analyze the test result to decide how to replace a defective cell of the memory on a spare line. The remedy analysis dedicated calculation unit (30) has a fail memory part (31) for storing a test result and a general-purpose remedy analysis part (32) for analyzing the test result according to an MRA program and for inserting, between the analysis process units, and executing a user function of a user analysis program.

(57) 要約: テストプロセサ (20) により半導体メモリを内蔵するDUTに試験信号を印加し、その応答信号に基づいてメモリの良否を判定し、救済解析専用計算ユニット (30) によりその試験結果を解析してメモリの欠陥セルをどのようにスペアラインで置き換えるかを決定する半導体デバイス試験装置であって、救済解析専用計算ユ

[続葉有]

WO 2005/078736 A1



BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

KZ, MD, RU, TJ, TM), ヨーロッパ (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

添付公開書類:

— 国際調査報告書

(84) 指定国 (表示のない限り、全ての種類の広域保護が可能): ARIPO (BW, GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), ユーラシア (AM, AZ, BY, KG,

2文字コード及び他の略語については、定期発行される各PCTガゼットの巻頭に掲載されている「コードと略語のガイダンスノート」を参照。

明細書

半導体デバイス試験装置及び試験方法

5 技術分野

この発明は、半導体デバイスが有しているメモリの不良メモリセルを予め準備しているスペアのメモリセルと置換して不良半導体デバイスを救済する半導体デバイス試験装置及び試験方法に関する。

10 従来の技術

集積回路として構成された半導体デバイスが有するメモリの不良メモリセルをスペアメモリセルにより効率的に救済する方法を、不良メモリセルの解析により求める装置をMRA（Memory Repair Analyzer：メモリ救済解析装置）と呼ぶ。ここで、半導体デバイスは専用のメモリデバイスのみならず、メモリデバイスを含むどのような半導体デバイスであってもよい。以下ではこの発明が対象とする半導体デバイスをメモリデバイスと呼ぶことにし、被試験メモリデバイスをDUT（Device Under Test）と表す。このMRAの手法はいくつかあるが、ここではその代表的な一つの例を以下に説明する。

図8Aは多数の半導体デバイスが配列形成されたウエハ110を示す。MRAはメモリ試験の前工程、つまりメモリデバイスのウエハ段階で行なわれる。ウエハ110には多数のメモリデバイスが配列形成されている。各メモリデバイスは複数のメモリブロックを有しており、各メモリブロックは多数のメモリセルで構成されている。図8Bに1つのメモリブロック120を模式的に示すように、各メモリブロック120に対しロー（Row:X軸方向）側とコラム（Column：Y軸方向）側とに複数のスペアライン130、135が用意されている。

いま、このメモリブロック120を試験した結果、メモリセル122と125と127とが不良であることが判明したとする。この試験結果のデータをMRAにより解析して救済解を求めることができる。図8Cでは、MRAによりメモリセル122のコラムライン131をコラム側のスペアライン130の1つに置き

換え、メモリセル125と127のローライン136をロー側のスペアライン135の1つに置き換える救済解を示している。このようなウエハに形成されたメモリデバイスの不良セルの救済については、例えば米国特許第6345004号や米国特許第6243307号に示されている。

- 5 このように、このMRAはメモリデバイスに発生した不良メモリセルを、どのスペアラインで置き換えるとリペア可能となるかを専用のハードウェアとソフトウェアとを用いて高速に解析することができるとする手段である。従来のメモリICの救済には十分に活動していた。

- 10 近年のメモリデバイスの発展はめざましく、64M-SDRAM以降、ユーザ（LSI製造メーカ）が指定した独自のリダンダンシイ（冗長）構造を有するメモリデバイスが増え、複雑化したことにより、後述のようにMRAの解析結果をポストプロセス（ウエハ段階における後工程）により調整しなければならないという問題が発生している。つまり、従来のMRA機能を使って救済解を求め、その後のポストプロセスでワークステーション（EWS：Engineering Work Station）
15 を用いて調整しなければならない。

- 20 しかしながら、この方法ではMRAの求めた1つの解しか調整することができないため、ポストプロセスで調整した結果が救済不可能となった場合に、ポストプロセスではリペアブルとなる解があるか否かを調べるができなかった。従って、多少の歩留まり低下が生じていた。このリダンダンシイ構造で従来のMRAが対応できなかったデバイスの一例を次に説明する。

- 25 図9に示すメモリデバイスは、4つのブロックグループBG1～BG4から成り、ブロックグループBG1とBG2でBANK-Aを、ブロックグループBG3とBG4でBANK-Bを構成している。また、各ブロックグループは4つのブロックから構成されている。ロー側のスペアライン135は各BANKに2本あり、2つのブロックグループを同時に救済する。コラム側のスペアライン130は各ブロックに2本ある。不良メモリセルに対する救済のためのスペアラインの割り当ての条件として、次の3つの条件が必要とされる。

図10に第1条件の説明図を示す。各ブロックのスペアライン2本は、同一ブロックグループ内のどの不良セルでも例外を除いて、原則的に自由に救済するこ

とができる。

図 1 1 に第 2 条件の説明図を示す。第 2 条件は、第 1 条件の例外であり、同一ブロックグループ内において、右端にあるブロック B L 4 のスペアラインは、左端にあるブロック B L 1 内の不良セルを救済することができない。同様に、左端にあるブロック B L 1 のスペアラインは、右端にあるブロック B L 4 内の不良セルを救済することができない

図 1 2 に第 3 条件の説明図を示す。これは同一 BANK 内において、隣接ブロックグループに属する互いに隣接するブロック、例えばブロックグループ B G 3 に属するブロック B L 4 とブロックグループ B G 4 に属するブロック B L 5 について、ブロック B L 4 で発生したアドレス a のフェイルを、このブロック B L 4 のスペアラインで救済したとする。その場合、隣接するブロック B L 5 では、同じアドレス a のフェイルをこのブロック B L 5 のスペアラインで救済することができない。ただし、その隣のブロック B L 6 のスペアラインでこのブロック B L 5 のアドレス a を救済することができるという条件である。

このように、図 9 の例はリダンダンシイ構造の 3 つの制約条件を有するメモリデバイスの例であるが、その他にも多種の構造のメモリデバイスがある。これらの多種類のメモリデバイスの救済を従来の M R A で全てを行なうことはできなくなってきた。例えば、前述の図 9 のメモリデバイスにおいては、第 1 条件と第 2 条件は従来の M R A で処理できたが、第 3 条件は対応できなかった。

前述したように、図 9 のリダンダンシイ構造のメモリデバイスでは、第 1 条件と第 2 条件とは従来の M R A で実行でき、M R A 実行後に、E W S 上で第 3 条件のチェックを行なうポストプロセスを実行して、救済結果を調整しなければならなかった。しかも、第 1 条件と第 2 条件の一つのみの救済解を得て、第 3 条件をチェックするので必ずしも最適解を求めることができず、歩留まり低下を生じることがあった。

つまり、図 1 3 の概念図に示すように、従来の M R A では、前述の一例の場合、D U T のファンクションテストを行い（ステップ S 1 4 0）、そのテスト結果のデータを入力してメモリ救済解析（M R A）を行い第 1 条件と第 2 条件の制限下で救済解を求め（ステップ S 1 4 1）、その救済解を一時保存し（ステップ S 1

4 2)、その結果と第3条件とをEWS上で調整し(ステップS 1 4 3)、最終救済解を出さなければならなかった。

本発明の目的は、汎用MR Aで第1条件から第3条件まで一括して解析し、救済可能な半導体デバイス試験装置及び試験方法を提供することである。

5

発明の開示

この発明による半導体デバイス試験装置は、

被試験半導体デバイスに試験信号を印加し、応答信号から不良メモリセルの情報を得るテストプロセサと、

10 上記不良メモリセル情報を救済解析し、不良メモリセルに対する救済方法を決める救済解析専用計算ユニット、

とを含み、上記救済解析専用計算ユニットは、

上記不良メモリセル情報をメモリ救済解析プログラムに従って救済解析を行い、不良メモリセルに対するスペアラインの割り当てを決めるメモリ救済解析手段と、

15 上記メモリ救済解析プログラムの所望の処理単位間にユーザ規定のユーザ解析プログラムに基づくユーザ関数を挿入し、上記メモリ救済解析プログラムによる処理データに変更を与えるユーザ関数手段、

とを含むように構成される。

この発明による半導体デバイスの試験方法は、以下のステップを含む：

20 (a) 被試験半導体デバイスのメモリに対しファンクションテストを行って不良メモリセルの情報を得るステップと、

(b) 上記不良メモリセル情報に対しメモリ救済解析を処理単位ごとに行い、不良メモリセルに対するスペアラインの割り当てを決めるステップと、

(c) 上記ステップ(b)における所望の処理単位間に、ユーザ規定の不良メモリセル救済条件に基づくユーザ関数を挿入し、上記メモリ救済解析プログラムによる
25 処理データに変更を与えるステップ。

図面の簡単な説明

図1は本発明の一実施例の構成概念図。

図 2 は図 1 における汎用救済解析部 3 2 の基本的な構成図。

図 3 は M R A プログラムと M R A 公開関数とユーザ解析プログラムの関係を示す概念図。

図 4 は M R A 公開関数の命名ルールを説明するための図。

5 図 5 はユーザ解析プログラムの例を示す図。

図 6 は本発明の他の実施例を示す機能構成図。

図 7 は本発明の更に他の実施例を示す機能構成図。

図 8 A は半導体デバイスが形成されたウエハを示す図。

図 8 B はメモリデバイス内の 1 つのブロックを示す図。

10 図 8 C は不良メモリセルの救済を説明するための図。

図 9 はリダンダンシイ構造のメモリデバイスの一例を示す図。

図 1 0 は図 9 のメモリデバイス救済の第 1 条件の説明図。

図 1 1 は図 9 のメモリデバイス救済の第 2 条件の説明図。

図 1 2 は図 9 のメモリデバイス救済の第 3 条件の説明図。

15 図 1 3 は従来技術の問題点を解決する目標の概念図である。

発明を実施するための最良の形態

発明の実施の形態を実施例に基づき図面を参照して説明する。

20 図 1 に本発明の一実施例の機能構成図を示す。図 1 に示すこの発明による半導体デバイス試験装置は、メモリを内蔵する半導体デバイス（以下では単にメモリデバイスと呼ぶ）を試験し、メモリデバイスに欠陥が有るときにはその場所等を解析し、そのメモリデバイスを救済するためにはメモリデバイスの不良ライン（メモリセル列又はメモリセル行）をどのスペアライン（スペアのメモリセルライン）で置換すればよいか最適な解を求めるものである。ここでは、その要点のみを説明することとする。

25 この半導体デバイス試験装置は、ワークステーション（EWS）1 0 と、テストプロセサ（TP）2 0 と、救済解析専用計算ユニット（RCPU）3 0 とから構成され、救済解析専用計算ユニット 3 0 にはフェイルメモリ部 3 1 と汎用救済解析部 3 2 とが設けられている。これらワークステーション 1 0、テストプロセ

サ 20、救済解析専用計算ユニット 30 はそれぞれの間で信号線 15、16、17 を介してデータ、制御信号、プログラムの送受信ができるようになっている。更に、DUT と電氣的に接触して DUT を試験するテストヘッド 40 がケーブル 18 を介してテストプロセサ 20 に接続されており、テストプロセサ 20 から試験信号を DUT に与え、DUT からの応答信号を得て試験結果をフェイルメモリ部 31 に書き込み、汎用救済解析部 32 によりフェイルメモリ部 31 に記憶されている試験結果を解析して DUT の不良メモリセルの救済をどのように行なうかを決定する。

ワークステーション (EWS) 10 は測定者が装置を操作するコンピュータであって、救済条件ファイル (RCF) 記憶部 11 と制御部 12 を有し、制御部 12 の制御の元に信号線 15、16、17、等を介して制御信号の送出、プログラムのダウンロード、データの送受信などを行なっている。その他、ワークステーション 10 はユーザが各種設定値、実行指示などを入力するための入力手段や、試験工程の表示、各種データの表示のための GUI (Graphical User Interface) としての表示装置を有しているが図示してない。

テストプロセサ (TP) 20 は、半導体デバイス試験装置専用に構成されたコンピュータであり、半導体デバイスを試験するテストプログラムを記憶するテストプログラム記憶部 21 を有し、DUT のファンクションテストのための制御等を行なう。テストプロセサ 20 はテストプログラム記憶部 21 に格納されたテストプログラムを実行することによりテストアドレス、テストデータ、期待値データを発生し、テストアドレスとテストデータをテストヘッド 40 に与え、テストヘッド 40 に装着されたウエハのメモリデバイスの上記テストアドレスにより指定されたメモリセルにテストデータを書き込む。次に、そのアドレスから読み出したデータを期待値データと比較してそのアドレスのメモリセルが良か不良か判定し、不良であれば救済解析専用計算ユニット 30 内のフェイルメモリ部 31 の対応するアドレスに不良を表すデータを書き込む。このようなテストをメモリデバイスの全アドレスについて実行することにより、テスト結果がフェイルメモリ部 31 に得られる。

救済解析専用計算ユニット 30 の汎用救済解析部 (汎用 MRA) 32 は、MR

Aプログラム記憶部32Aとユーザ解析プログラム記憶部32Bと、解析制御部32Cと、データ記憶部32Dとを有し、DUTに対する試験結果のデータを記憶しているフェイルメモリ部31から必要なデータを取得してMRAプログラムを解析制御部32Cの制御の下に実行して不良セルの救済解析を行ない、不良メモリセルに対する最も効率的なスペアラインの割り当てを決定する。この解析結果はテストプロセサ20に送られ、後でそのDUTの物理的リペアを行うときに使用される。ユーザ解析プログラム記憶部32B及びデータ記憶部32Dについては以下に図2を参照して説明する。

図2は、図1における救済解析専用計算ユニット30に内蔵されている汎用救済解析部32の基本的な機能構成図である。この発明の基本となる汎用救済解析部32は、図1を参照して簡単に述べたようにメモリ救済解析(MRA)プログラム32APを記憶するMRAプログラム記憶部32Aと、ユーザ解析プログラム関数32BPを記憶するユーザ解析プログラム記憶部32Bと、解析制御部32Cと、データ記憶部32Dとで構成されている。MRAプログラム記憶部32Aに記憶されるMRAプログラム32APは、従来のメモリ救済解決のための定型の垂れ流しプログラムではなく、解析プロセスを一動作毎にパーツ化し、そのパーツ化された一動作の終了点でユーザ専用のDUTの救済解析を行なうユーザ解析プログラム32BP中のユーザ関数32B1~32B5を挿入できるユーザ関数挿入ポイント32N1~32N5が設けられている。データ記憶部32DにはMRAプログラムの実行に必要なデータ、例えばリペア条件ファイル、解析の中間段階データ、解析結果のデータを保存する。

MRAプログラム32APは、例えばテストプロセサ20からのDUTに対する試験終了トリガを受けてフェイルメモリ部31からテスト結果のデータを取得し、解析用変数を初期化するステップ32A1と、テスト結果のデータを解析して不良メモリセルラインに対し救済するスペアラインの割り当てを決めるステップ32A2と、ステップ32A2後の残りの不良セルに対し、救済するスペアラインの割り当てを決めるステップ32A3と、ビット救済後、スペアラインの他の割り当て方法があるか判定するステップ32A4と、他の割り当て方法がなければ、解析結果を作成するステップ32A5とを有し、ステップ32A4で他の割り当て方法があれば

サイドステップ 32A3 のビットフェイル救済のためのスペアライン割り当てを行う。各メモリデバイスに対しこのようなステップ 32A1～32A5 を実行する。ステップ 32A2 のラインフェイル救済、ステップ 32A3 のビットフェイル救済、ステップ 32A4 の判定などにおいては DUT のメモリサイズ、各ブロックグループを構成するブロック数、ロー用スペアライン数、コラム用スペアライン数などの情報
5 が必要であり、これらの情報はリペア条件ファイル RCF からデータ記憶部 32D にロードされているものを使用する。

MRA プログラム 32AP における解析プロセスの一動作毎のパーツ 32 化とは、例えば、テスト結果のデータ取得および解析用変数の初期化（ステップ 32A1）や、
10 ラインフェイルの解析（ステップ 32A2）や、ビットフェイルの解析（ステップ 32A3）や、救済結果の作成（ステップ 32A5）等に分けて処理を区分することを言う。この各処理と処理の間に必要に応じてユーザ関数 32B1～32B5 を授受する挿入ポイント 32N1～32N5 を設け、ユーザ専用のユーザ解析プログラム 32BP を構成する個々のユーザ関数 32B1～32B5 を挿入できるようにする。

15 ユーザ関数としては、例えば、変数初期化時に実行する関数 32B1 やラインフェイル救済後に実行する関数 32B2、ビットフェイル救済後に実行する関数 32B3、結果作成後に実行する関数 32B4、結果作成後に実行する関数 32B5 等がある。これらは解析制御部 32C の制御の下に実行される。具体的には、ラインフェイル救済後に実行する関数は、例えば救済アドレスに同じものがないかチェックする
20 関数である。ビットフェイル救済後に実行する関数もここでは救済アドレスに同じものがないかチェックする関数である。このチェックにより、図 12 で説明した同一アドレスに対する救済に使用すべきスペアラインの正しい選択を行うことができる。

このようにユーザは挿入するユーザ関数を用いて MRA プログラムに従った処理の所望の段階で解析データ、つまり救済情報やフェイル情報などを得ることにより、特殊なリダンダンシイ構造の DUT をも救済できる。図 1 及び 2 で説明した実施例において、解析制御部 32C と MRA プログラム記憶部 32A との組はフェイル情報をメモリ救済解析するメモリ救済解析手段を構成し、解析制御部 32C とユーザ解析プログラム記憶部 32B との組は MAR プログラムの処理単位
25

間にユーザ関数を挿入するユーザ関数手段を構成している。

図3は図1及び2に示した実施例の変形実施例として、図2に示したMRAプログラム32APとユーザ解析プログラム32BP中のユーザ関数とがユーザに直接的に理解しやすいMRA公開関数32Fを介在して情報交換する機能を模式的に示す。MRAプログラム32APが使用するデータ記憶部32Dにユーザ関数32B1〜32B5から直接的にアクセスしてデータの設定をすると、ユーザ関数についてのユーザの誤った値の設定により救済データ等を壊す恐れがある。そこで、MRAプログラム32APとユーザ関数32B1〜32B5との間にMRA公開関数と名付けたフィルターとしての専用の関数32Fを設ける。このMRA公開関数32Fを使うことで、ユーザ関数32B1〜32B5内でのデータベースの参照や変更を安全に行なうことができる。ユーザ関数はこの実施例ではC言語を用いて記述されている。

図3で説明したMRA公開関数32Fは、半導体デバイス試験装置のメーカ技術者やユーザ技術者が、ひんばんに使用することを想定して、一定のルールを規定し、ユーザがユーザ解析プログラムを容易に作成できるようにする。MRA公開関数名から意味がおおよそ解ること、ユーザ関数作成時のストレスを軽減させること、ユーザに使いづらい、という印象を与えないこと等を考慮して、図4に示すようにMRA公開関数名を規定する。

図4のMRA公開関数名は、その一例であり、一定のルールで記述するようにしたものでその内容を説明する。まず、全関数共通に“Mra”というタグを付ける。次にどの種類のデータを扱うかという“クラス名”を付ける。例えば、救済解析のデータ記憶部32Dに保存されている解析結果情報を扱うのであれば“Result”とし、欠陥情報であれば“Fail”とし、救済情報を扱うのであれば“Repair”というクラス名にする。

次に何をするかという動詞を付ける。例えば情報を得るのであれば“Get”、データをセットするのであれば“Set”とする。そして、最後にどの情報を扱うかという目的語をおく。一例を挙げると、“A=MraResultGetTotalBin”のような記述になるが、プログラム作成上の決め事であるので種々な記述が考えられる。要は、誰でも何時でも容易に記述できるようにしたものである。この実施例では更に、ユーザ関数によるデータの設定に対し、異常データでないかチェックするデータチェック

関数部 3 2 F C が設けられている。データチェック関数部 3 2 F C はユーザ関数 32B1～32B5 により入力された設定データ、例えばアドレス値、スペアライン数、スペアグループ番号、ブロック番号等に異常な値がないか、即ち、それぞれ予め決めた範囲内の値となっているかをチェックし、異常な値を検出すると設定エラーの表示をワークステーションに表示する。

図 5 は図 4 の記述法を取り入れたユーザ解析プログラムの一例を示す。このユーザ解析プログラム 32BP は、ワークステーション 1 0 からユーザ解析プログラム記憶部 3 2 B にロードされる。行 2～1 3 (U F 1) は救済アドレス部分のソートを実行するユーザ関数であり、行 1 4～2 9 (U F 2) は救済アドレスに同じものがないかをチェックする関数であり、行 3 0～3 6 (U F 3) はユーザ解析プログラムのメイン関数であり、行 3 7～4 4 (U F 4) はユーザ・セットアップ関数である。M R A 公開関数の記述法は例えば行 8、9、2 4、2 5 などに見られる。

例えば図 2 の M R A プログラム 32AP において、ユーザ関数挿入ポイント 32N2 15 では、メイン関数 U F 3 が実行される。図 5 のプログラムにおける行 3 0 の "SampleRuleCheck1" はユーザ関数名を表す。従って、行 3 4 の "addressCheckRow" によるチェックはユーザ関数 U F 2 の行 1 4 から 2 9 により実行される。行 3 9 の "MraPOINT_POST_REPAIR_LINEFAIL" はユーザ関数挿入ポイント 32N2 につけられた名称である。

ユーザ関数 U F 2 における行 2 0 は図 4 に示した記述法を使用し、M R A 公開関数 "MraBlockGroupGetRepairList" を表し、そのリスト中のユーザが設定したブロックグループ数 "blockGroupNo" 及びロー又はコラムの指定 "dir" は、この関数を実行するときに設定値が予測される値、例えば blockGroupNo は 8 個以下、dir は 1 (row) 又は 2 (column)、となっているかチェックすることにより、ユーザが誤って 25 とり得ない値を設定した場合にそれを検出してエラーメッセージを表示することができる。誤った値をデータ記憶部 3 2 D に設定することを避けることができる。

図 6 は図 4 及び 5 による M R A 公開関数を導入した実施例の構成を示す。ただし、図を簡略化するため、テストプロセサ 2 0、テストヘッド 4 0 などは図示していない。この実施例では、図 1 の構成に対し、更に M R A 公開関数記憶部 3 2 F

Cが汎用救済解析部32Bに設けられている。試験開始前に予めワークステーション10からMRAプログラム32AP、ユーザ解析プログラム32BP、及びMRA公開関数32FをそれぞれMRAプログラム記憶部32A、ユーザ解析プログラム記憶部32B、及びMRA公開関数記憶部32FCにロードしておく。解析制御部32CとMRA公開関数記憶部32FCの組は、ユーザ関数をMRA公開関数を介してMRAプログラムの処理単位間に挿入するためのMRA公開関数手段を構成している。

図7は図1の実施例において複数種類のDUTに対してユーザ関数を切り替え可能にした実施例の汎用救済解析部32の構成を示す。リペア条件ファイルRCFは半導体デバイスの種類ごとに異なっており、従ってDUTの種類を切り替えるごとに対応するRCFを選択してMRAプログラムを実行しなければならない。DUTの品種切り替えが頻繁に行なわれる場合に、その都度ワークステーション10から汎用救済解析部32にロードするのでは時間がかかり、効率が悪い。そこで、DUTに応じたユーザ関数32BJ、32BKの切り替えを容易に行なえるように、ユーザ解析プログラム32BPには複数種類のDUTに対応したユーザ関数の組32BJ及び32BKとが設けられ、ユーザ解析プログラム記憶部32Bに格納される。これらユーザ関数の組にはルール名"J"及び"K"がそれぞれ付けられている。

汎用救済解析部32には更にRCF記憶部32Fが設けられ、その中にDUTの種類に依存しないルール名"none"のリペア条件ファイルRCF1、DUTの種類に依存するルール名"J"及び"K"のリペア条件ファイルRCF2、RCF3が設けられている。ルール名"J"のRCFは例えば容量128Mバイトのメモリデバイス用RCFであり、ルール名"K"のRCFは容量256Mバイトのメモリデバイス用RCFである。このルール名がDUTの識別ともなるため、ユーザが容易にユーザ関数を定義できるようになる。

ワークステーション10の救済条件ファイル記憶部11には、前述したように救済条件ファイル(RCF)が格納されており、このファイルを基に救済解析を行なう。このRCFにルール名を指定することで、実行時にDUTの種類に応じてユーザ関数を容易に換えられるようにする。

RCFにルール名が指定されていなければ、MRAプログラム32APはオリジ

ナルな救済解析を行なう。図 7 に示すように R C F にルール名が指定されていれば、そのルール名 "K" に指定されたユーザ関数の組 32BK が実行される。例えば図 5 のユーザ解析プログラムにおいて、行 3 9 及び 4 1 の "SAMPLE_RULE_CHECK1" はリペア条件ファイル R C F 2 のルール名 "J" (又は R C F 3 の "K") に相当する。

このように、一台の半導体デバイス試験装置で異なる D U T を測定する場合でも D U T に応じたユーザ関数を簡単に切り替えることができる。

発明の効果

- 10 以上詳細に説明したように、従来の M R A では従来のメモリ I C の救済には充分にその能力を発揮していた。ところが、近年のユーザ独自の特殊なリダンダンシイ構造によるメモリデバイスを内蔵する D U T の救済が困難になってきた。そして、イーールドダウン (歩留まり低下) が発生するようになってきた。

- この発明によると、新たに設けた M R A プログラム 32AP とユーザ解析プログラム 32BP とを連携させた汎用救済解析部 3 2 とを設けて、特殊なリダンダンシイ構造の D U T にも対応できるようにした。その結果、ウエハ段階でのポストプロセスが不要となった。更に、ユーザ専用処理であるユーザ関数が組み込めるので、ユーザ専用処理が救済可能と判断するまで救済解析をやり直すことができるようになった。

- 20 ユーザ専用処理が救済解析専用計算ユニット (R C P U) 3 0 上で動作するので、多数個同時試験の際にはテストプロセサ 2 0 による D U T のファンクションテストと救済解析専用計算ユニット 3 0 によるメモリ救済解析の並列処理が可能となり、実行時間をかなり短縮することができた。

- このようにこの発明によると、半導体メモリデバイスを含む D U T の救済解析
25 がほぼ完全にできるようになり、その技術的效果は大である。

請求の範囲

1. 被試験半導体デバイスに試験信号を印加し、応答信号から不良メモリセルの情報を得るテストプロセサと、
- 5 上記不良メモリセル情報を救済解析し、不良メモリセルに対する救済方法を決める救済解析専用計算ユニット、
とを含み、上記救済解析専用計算ユニットは、
上記不良メモリセル情報をメモリ救済解析プログラムに従って救済解析を行い、不良メモリセルに対するスペアラインの割り当てを決めるメモリ救済解析手段と、
- 10 上記メモリ救済解析プログラムの所望の処理単位間にユーザ規定のユーザ解析プログラムに基づくユーザ関数を挿入し、上記メモリ救済解析プログラムによる処理データに変更を与えるユーザ関数手段、
とを含む半導体デバイス試験装置。
2. 請求項1記載の半導体デバイス試験装置において、上記救済解析専用計算ユ
15 ニットは、上記ユーザ関数をメモリ救済解析公開関数を介して上記メモリ救済解析プログラムの所望の処理単位間に挿入するメモリ救済解析公開関数手段を含んでいる。
3. 請求項2記載の半導体デバイス試験装置において、上記メモリ救済解析公開関数は上記ユーザ関数により設定されるデータが正常であることをチェックするデ
20 ータチェック関数部を有している。
4. 請求項1記載の半導体デバイス試験装置において、上記救済解析専用計算ユ
ニットは、
上記テストプロセサからの上記不良メモリセル情報を格納するフェイルメモリ
部と、
- 25 上記メモリ救済解析プログラムを格納するメモリ救済解析プログラム記憶部と、
上記ユーザ解析プログラムを格納するユーザ解析プログラム記憶部と、
上記メモリ救済解析プログラムの実行と上記ユーザ解析プログラムの実行を制御する解析制御部、
とを含み、上記解析制御部と上記メモリ救済解析プログラム記憶部とは上記メモ

リ救済解析手段を構成し、上記解析制御部と上記ユーザ解析プログラム記憶部とは上記ユーザ関数手段を構成している。

5. 請求項4記載の半導体デバイス試験装置において、上記救済解析専用計算ユニットは、半導体デバイスの種類ごとにその救済条件を規定した複数の救済条件

- 5 ファイルを記憶する救済条件ファイル記憶部を有し、上記ユーザ解析プログラム記憶部には上記複数の救済条件ファイルに対応して規定された複数組のユーザ関数が上記ユーザ解析プログラムとして記憶されており、上記解析制御部は上記被試験半導体デバイスの種類と一致する救済条件ファイルに基づいてユーザ関数の組を選択して上記メモリ救済解析プログラムの処理単位間に挿入する。

- 10 6. 半導体デバイスの試験方法であり、以下のステップを含む：

(a) 被試験半導体デバイスのメモリに対しファンクションテストを行って不良メモリセルの情報を得るステップと、

(b) 上記不良メモリセル情報に対しメモリ救済解析を処理単位ごとに行い、不良メモリセルに対するスペアラインの割り当てを決めるステップと、

- 15 (c) 上記ステップ(b)における所望の処理単位間に、ユーザ規定の不良メモリセル救済条件に基づくユーザ関数を挿入し、上記メモリ救済解析プログラムによる処理データに変更を与えるステップ。

7. 請求項6記載の半導体デバイス試験方法において、上記ステップ(c)は上記ユーザ関数をメモリ救済解析公開関数を介在させて上記メモリ救済解析プログラムの

- 20 の処理単位間に挿入するステップを含む。

8. 請求項7記載の半導体デバイス試験方法において、上記メモリ救済解析公開関数は、上記ユーザ関数により設定されるデータが正常であるかチェックするデータチェック関数を実行するステップを含む。

- 25 9. 請求項6記載の半導体デバイス試験方法において、上記メモリ救済解析プログラムは、ラインフェイル救済処理ステップと、ビット救済処理ステップとを含み、上記ステップ(c)は上記ラインフェイル救済処理ステップの次及び上記ビット救済処理ステップの次にそれらの処理結果に上記ユーザ関数によりそれぞれ変更を与えるステップを含む。

10. 請求項6記載の半導体デバイス試験方法において、上記ステップ(c)は、

半導体デバイスの種類に対応して予め決められた複数の救済条件に対応して複数の組のユーザ関数が設けられており、上記被試験半導体デバイスの種類に対応したユーザ関数の組を選択し、上記メモリ救済解析プログラムの処理単位間に挿入するステップを含む。

1/10

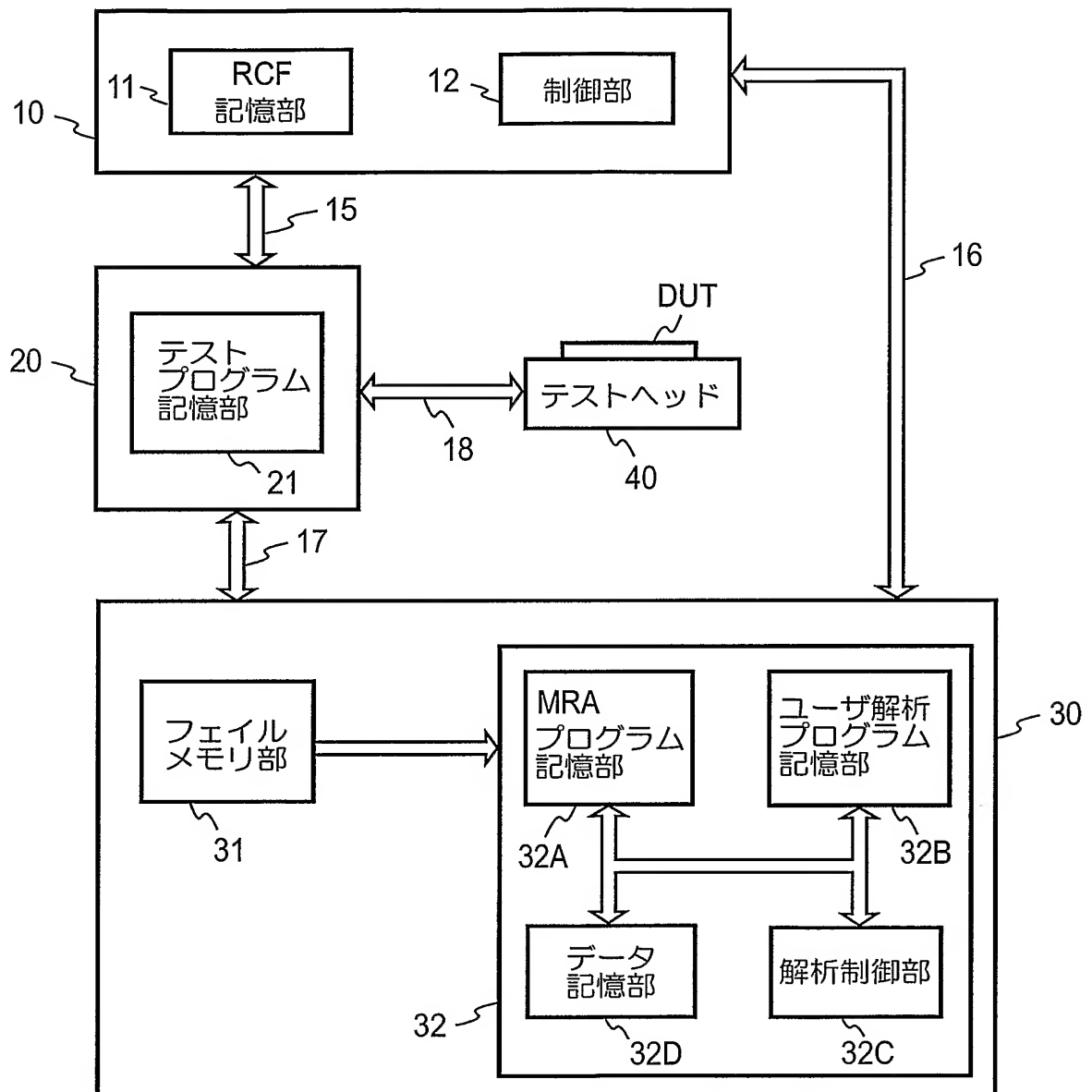


図 1

2/10

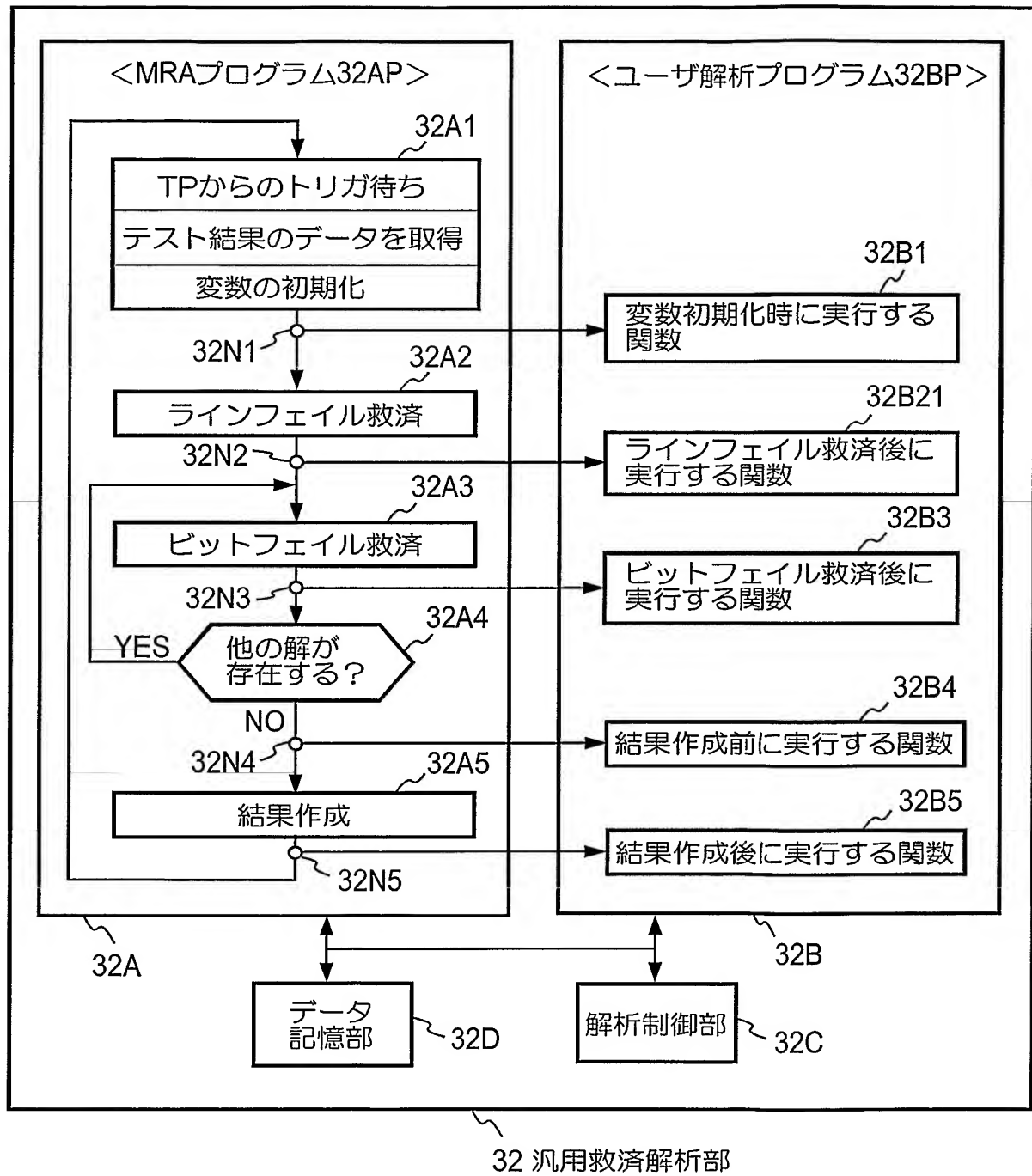


図2

3/10

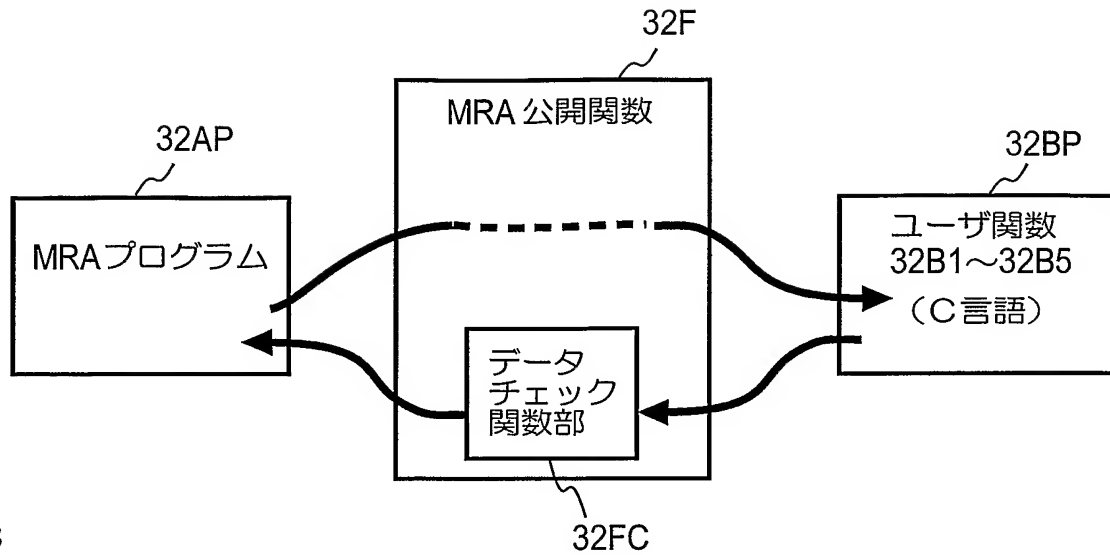


図 3

MRA公開関数：“Mra” + クラス名 + 動詞 + 目的語

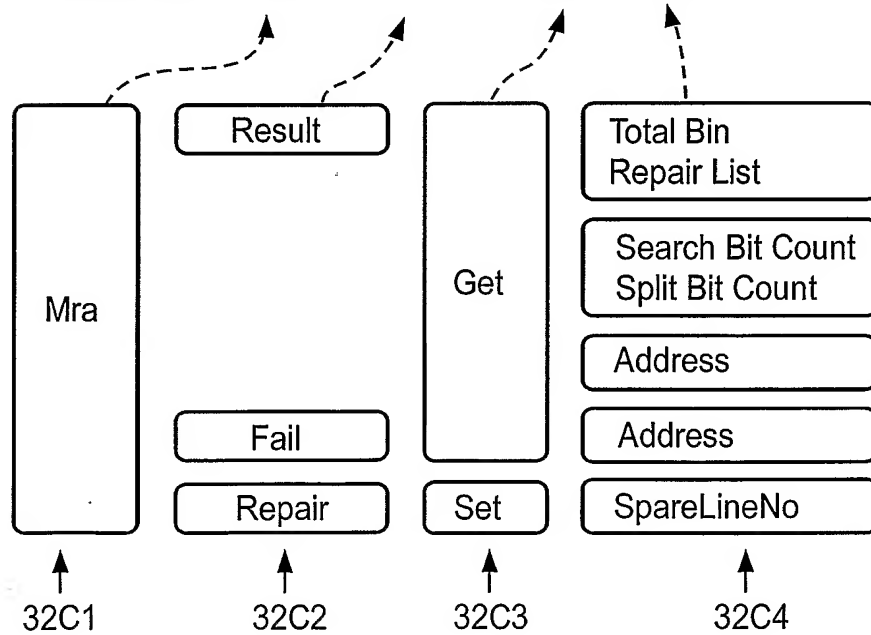


図 4

4/10

図5

```

1  #include <MraLib.h>

2  static  MraRepairId_t  RepairIdList[MraSPARELINE_COUNT_MAX];
3  static  int            compare( const void *a, const void *b)
4  {
5      const MraRepairId_t  *A = a, *B = b;
6      int  blockA, blockB;
7      uint32_t      adrsA, adrsB;
8      MraRepairGetAddress( *A, &blockA, &adrsA);
9      MraRepairGetAddress( *B, &blockB, &adrsB);
10     if (  adrsA < adrsB)          return  -1 ;
11     if (  adrsA > adrsB)          return   1 ;
12     return                      0 ;
13 }

14 static  MraStatus_t addressCheckRow(int blockGroupNo, MraDir_t dir)
15 {
16     MraRepairId_t *repairId = RepairIdList;
17     int            blockNo0, blockNo1 ;
18     uint32_t      address0, address1;
19     int i, count ;
20     count = MraBlockGroupGetRepairList ( blockGroupNo, dir,
21                                           repairId, MraSPARELINE_COUNT_MAX);
22     MraQsort( repairId, count, sizeof(MraRepairId_t), compare);
23     for(i=0; i<count-1; i++) {
24         MraRepairGetAddress( repairId[ i ], &blockNo0, &address0);
25         MraRepairGetAddress( repairId[ i+1], &blockNo1, &address1);
26         if ( address0 == address1) return MraNG ;
27     }
28     return MraOK ;
29 }

30 MraStatus_t SampleRuleCheck1(void *clientdata)
31 {
32     int blockGroupNo ;
33     blockGroupNo = MraGetBlockGroupNo( ) ;
34     if ( addressCheckRow ( blockGroupNo, MraDIR_ROW) != MraOK) return MraNG ;
35     return MraOK ;
36 }

37 int      UserSetup ( void )
38 {
39     MraSetUserFunc("SAMPLE_RULE_CHECK1", MraPOINT_POST_REPAIR
40                  _LINEFAIL, SampleRuleCheck1,
41                  NULL ) ;
42     MraSetUserFunc("SAMPLE_RULE_CHECK1", MraPOINT_POST_REPAIR
43                  _BITFAIL, SampleRuleCheck1,
44                  NULL);
45     return 0 ;
46 }

```

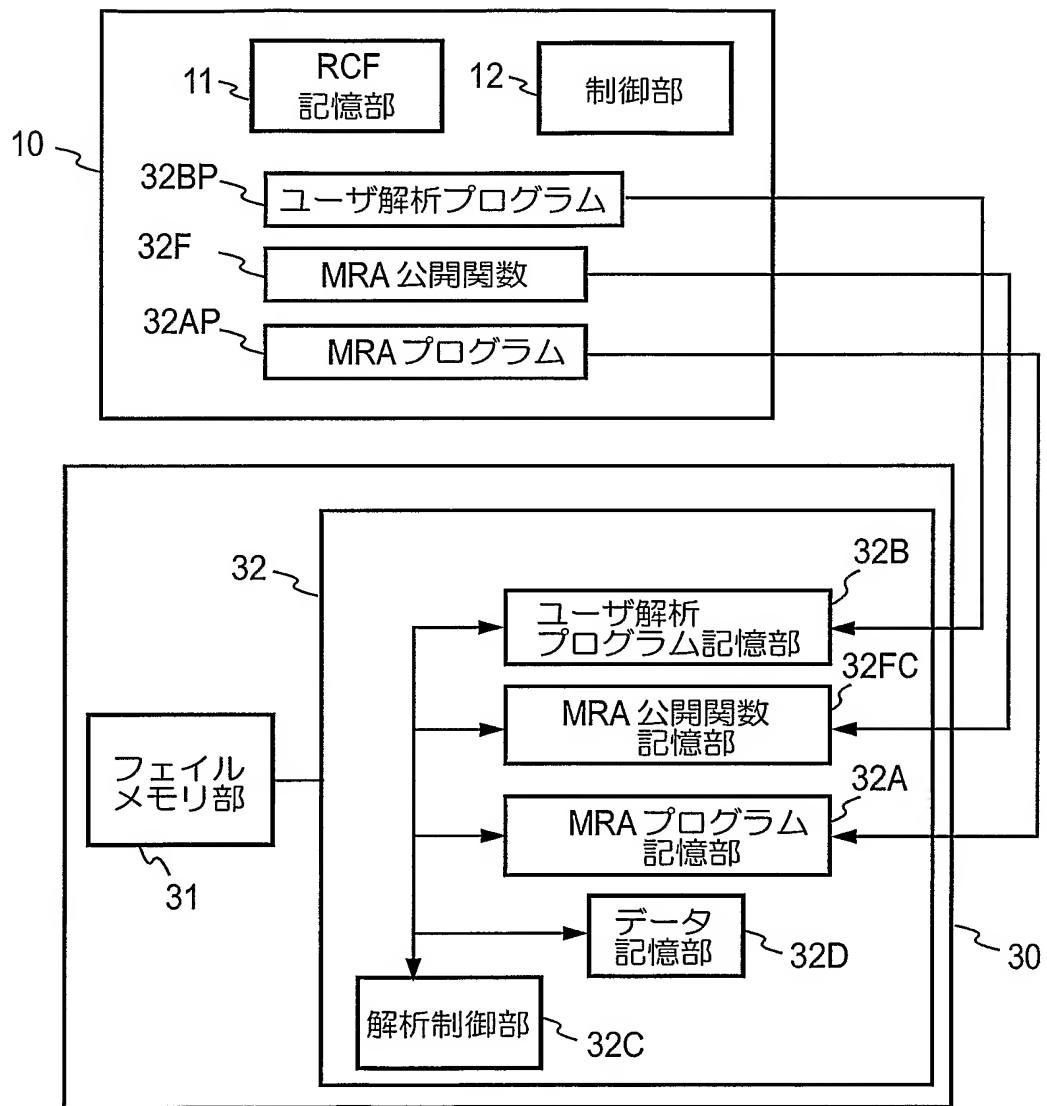
UF1

UF2

UF3

UF4

図6



6/10

図7

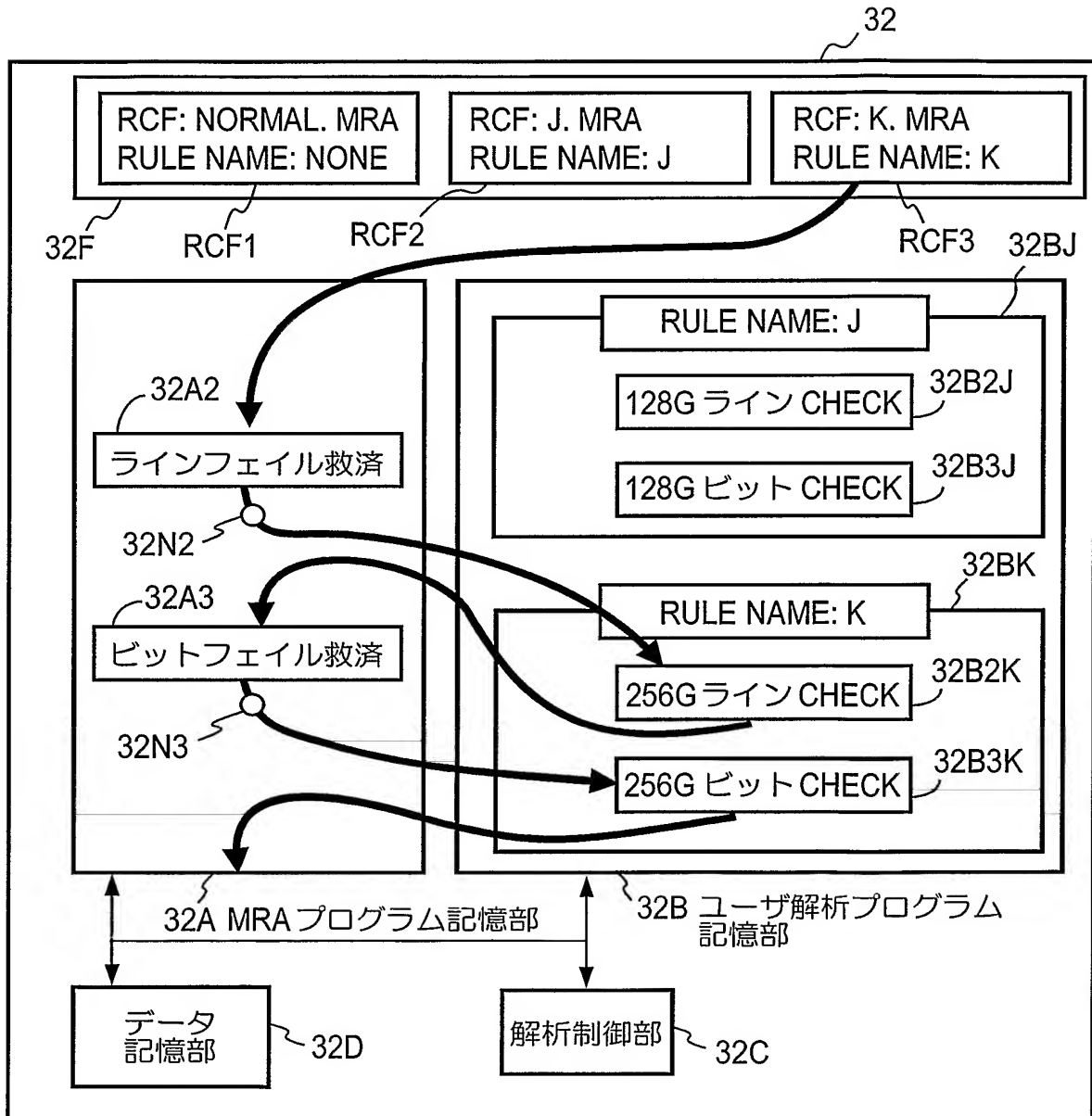


図8A

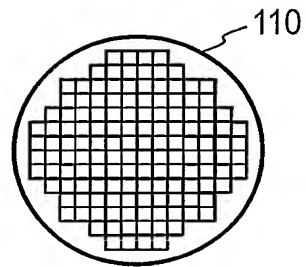


図8B

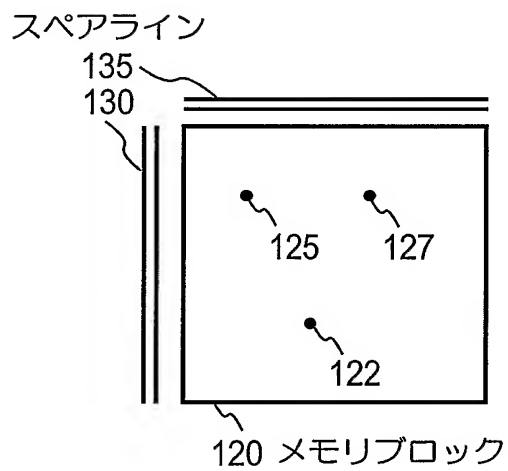


図8C

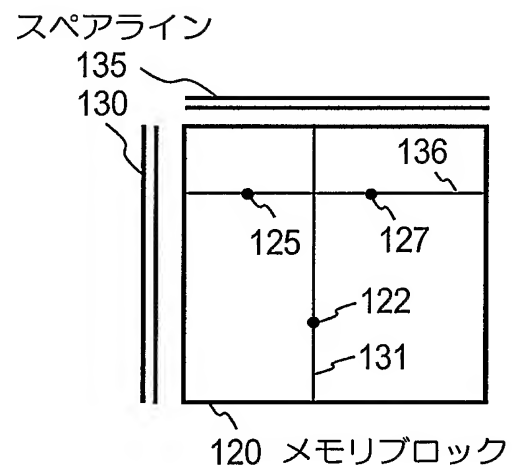


図9

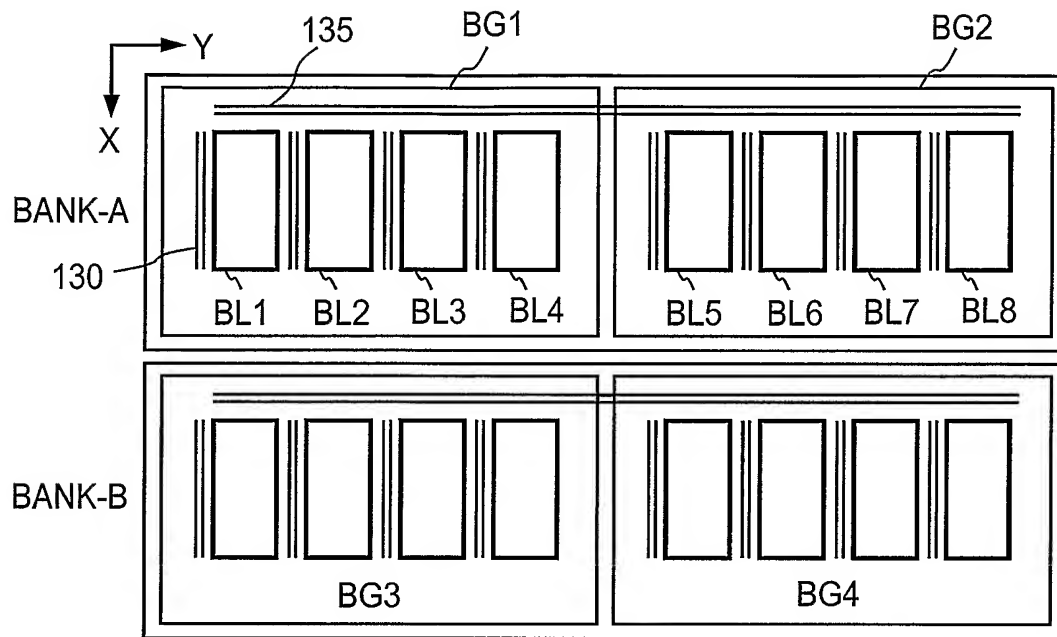
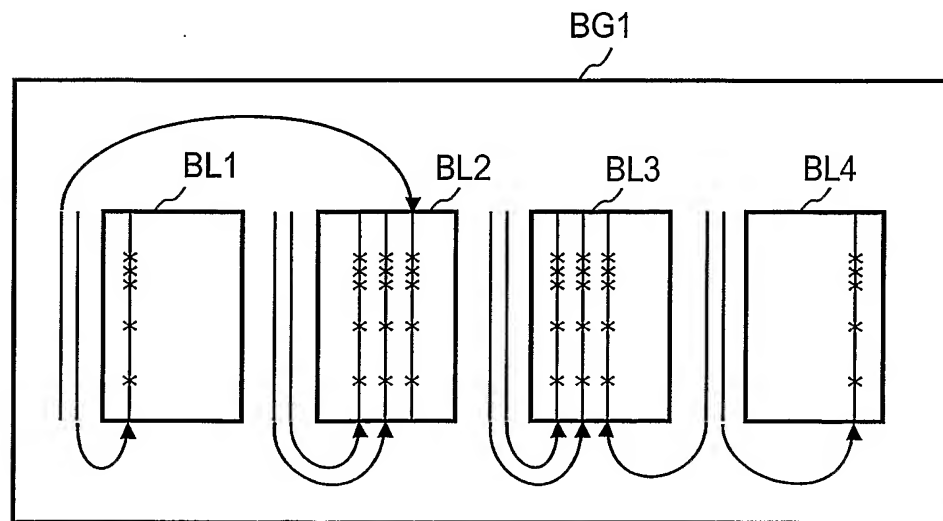


図10



9/10

図11

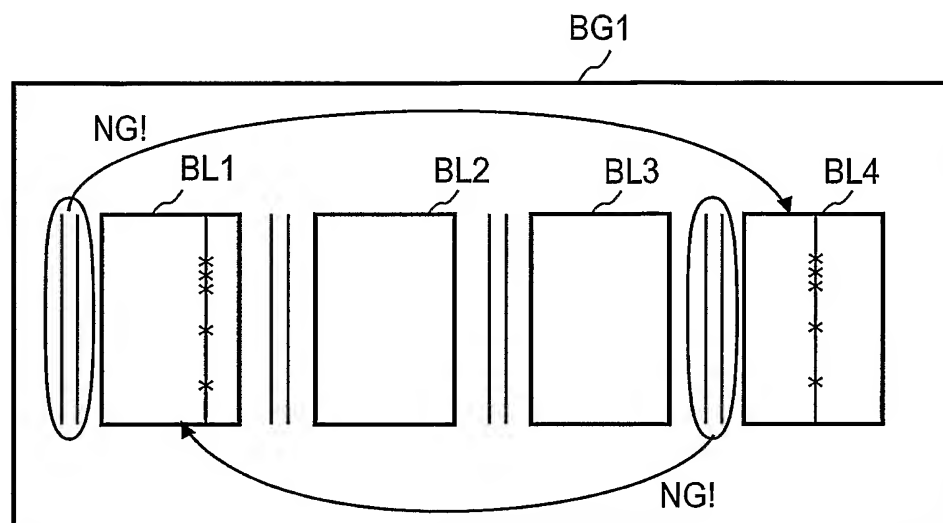
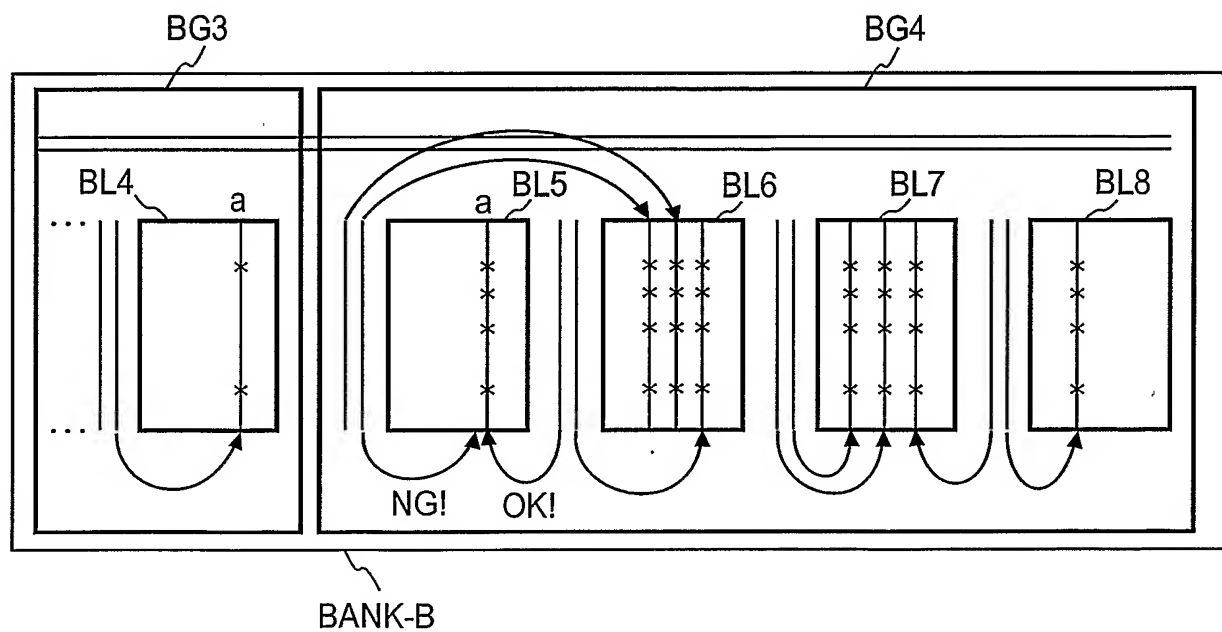
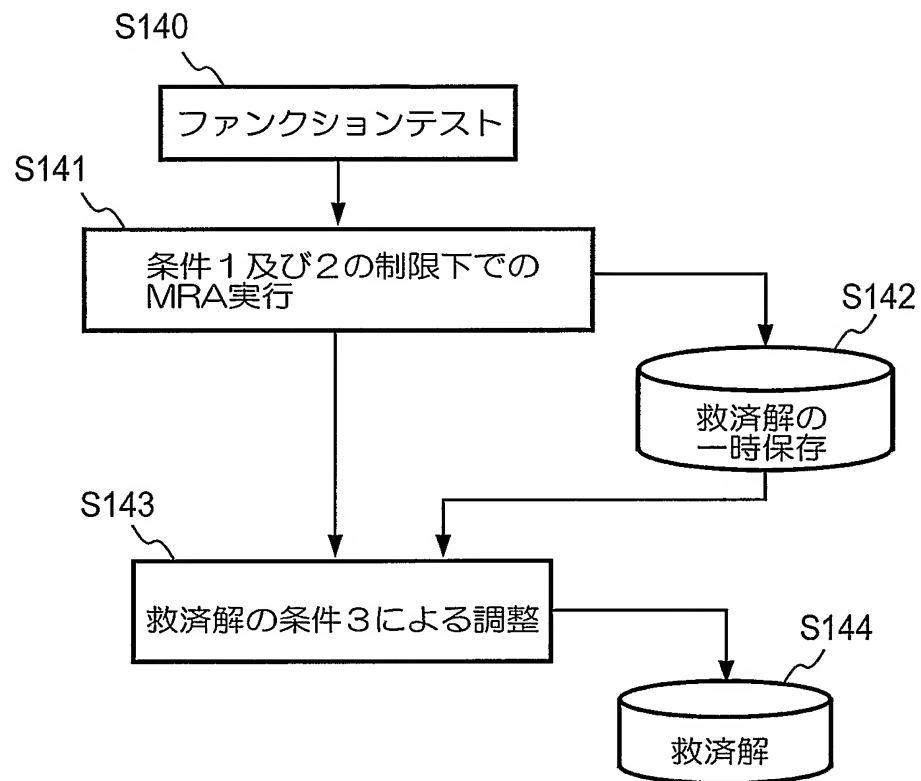


図12



10/10

図13



INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2004/001805

A. CLASSIFICATION OF SUBJECT MATTER

Int.Cl⁷ G11C29/00, G01R31/28

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

Int.Cl⁷ G11C29/00, G01R31/28

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

| | | | |
|---------------------------|-----------|----------------------------|-----------|
| Jitsuyo Shinan Koho | 1922-1996 | Jitsuyo Shinan Toroku Koho | 1996-2004 |
| Kokai Jitsuyo Shinan Koho | 1971-2004 | Toroku Jitsuyo Shinan Koho | 1994-2004 |

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|-----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| Y | JP 11-213695 A (Advantest Corp.), 06 August, 1999 (06.08.99), Full text; all drawings (Family: none) | 1-10 |
| Y | Richard M. Stallman, translated by Toru TAKIZAWA, "GNU Emacs 19 Manual", Addison Wesley Publishers Japan Kabushiki Kaisha, 31 October, 1997 (31.10.97), pages 337 to 338 | 1-10 |
| Y | JP 2002-216495 A (Mitsubishi Electric Corp.), 02 August, 2002 (02.08.02), Full text; all drawings & US 2002/0095630 A1 | 5, 10 |

☐ Further documents are listed in the continuation of Box C.

☐ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search
18 November, 2004 (18.11.04)

Date of mailing of the international search report
21 December, 2004 (21.12.04)

Name and mailing address of the ISA/
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

A. 発明の属する分野の分類 (国際特許分類 (IPC))

Int. Cl⁷ G11C29/00, G01R31/28

B. 調査を行った分野

調査を行った最小限資料 (国際特許分類 (IPC))

Int. Cl⁷ G11C29/00, G01R31/28

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報 1922-1996年

日本国公開実用新案公報 1971-2004年

日本国実用新案登録公報 1996-2004年

日本国登録実用新案公報 1994-2004年

国際調査で使用した電子データベース (データベースの名称、調査に使用した用語)

C. 関連すると認められる文献

| 引用文献の カテゴリー* | 引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示 | 関連する 請求の範囲の番号 |
|-----------------|-------------------------------------------------------------------------------------------------|------------------|
| Y | JP 11-213695 A (株式会社アドバンテスト) 1999. 08. 06, 全文, 全図 (ファミリーなし) | 1-10 |
| Y | リチャード・M・ストールマン著, 滝沢徹訳, "GNU Emacs19マニユアル", アジソン・ウェスレイ・パブリッシャーズ・ジャパン株式会社, 1997. 10. 31, p337-338 | 1-10 |
| Y | JP 2002-216495 A (三菱電機株式会社) 2002. 08. 02, 全文, 全図 & US 2002/0095630 A1 | 5, 10 |

☐ C欄の続きにも文献が列挙されている。☐ パテントファミリーに関する別紙を参照。

* 引用文献のカテゴリー

「A」特に関連のある文献ではなく、一般的技術水準を示すもの

「E」国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの

「L」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献 (理由を付す)

「O」口頭による開示、使用、展示等に言及する文献

「P」国際出願日前で、かつ優先権の主張の基礎となる出願

の日の後に公表された文献

「T」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの

「X」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの

「Y」特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの

「&」同一パテントファミリー文献

国際調査を完了した日

18. 11. 2004

国際調査報告の発送日

21.12.2004

国際調査機関の名称及びあて先

日本国特許庁 (ISA/JP)

郵便番号100-8915

東京都千代田区霞が関三丁目4番3号

特許庁審査官 (権限のある職員)

加藤 俊哉

5N

9554

電話番号 03-3581-1101 内線 3546